

(12) INTERNATIONAL APPLICATION PUBLISHED UNDER THE PATENT COOPERATION TREATY (PCT)

(19) World Intellectual Property Organization
International Bureau



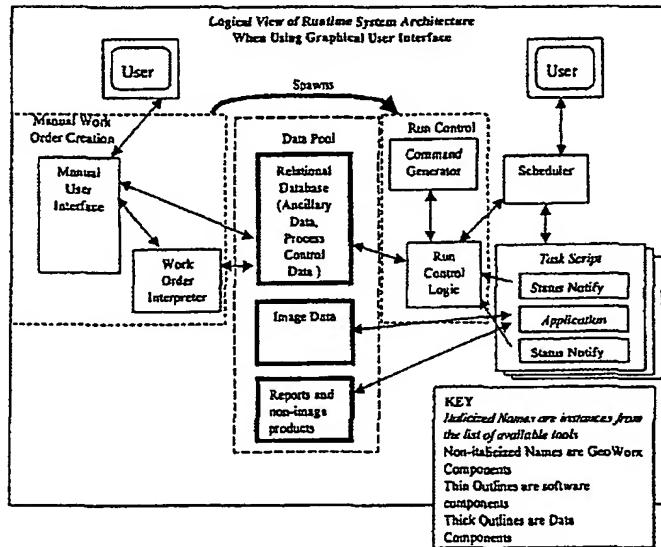
(43) International Publication Date
30 August 2001 (30.08.2001)

PCT

(10) International Publication Number
WO 01/63529 A1

- (51) International Patent Classification⁷: G06F 17/60 (81) Designated States (*national*): AE, AG, AL, AM, AT, AU, AZ, BA, BB, BG, BR, BY, BZ, CA, CH, CN, CR, CU, CZ, DE, DK, DM, DZ, EE, ES, FI, GB, GD, GE, GH, GM, HR, HU, ID, IL, IN, IS, JP, KE, KG, KP, KR, KZ, LC, LK, LR, LS, LT, LU, LV, MA, MD, MG, MK, MN, MW, MX, MZ, NO, NZ, PL, PT, RO, RU, SD, SE, SG, SI, SK, SL, TJ, TM, TR, TT, TZ, UA, UG, UZ, VN, YU, ZA, ZW.
- (21) International Application Number: PCT/US01/05712
- (22) International Filing Date: 23 February 2001 (23.02.2001)
- (25) Filing Language: English
- (26) Publication Language: English
- (30) Priority Data:
60/184,570 24 February 2000 (24.02.2000) US
09/791,124 22 February 2001 (22.02.2001) US
- (71) Applicant: VERIDIAN ERIM INTERNATIONAL [US/US]; 3300 Plymouth Road, Ann Arbor, MI 48105 (US).
- (72) Inventors: SIERON, Russell, J.; 27600 Western Golf Ct., Livonia, MI (US). MOREY, Bruce @; 3463 Yellowstone, Ann Arbor, MI (US).
- (74) Agents: POSA, John, G. et al.; Gifford, Krass, Groh, Sprinkle, Anderson & Cikowski, PC, Suite 400, 280 N. Old Woodward Avenue, Birmingham, MI 48009 (US).

(54) Title: WORKFLOW SYSTEM AND BUILDER OFFERS IMAGE SCRIPT TOOLS ON INPUT DATA TYPE



WO 01/63529 A1

(57) Abstract: A Process Automation System (figure 1) and accompanying method provides for the automated or semi-automated creation of workflow management procedures having an arbitrary number of tasks to be performed manually or via computer automation in specified sequences. Workflows are automatically or semi-automatically generated by prompting the user with valid choices. After creating the workflow, the system then acts as the executive for execution; initiating tasks, tracking status, and managing intermediate and final products. The capability is provided to define new products and tasks for the system via a graphical user interface.

WORKFLOW SYSTEM AND BUILDER OFFERS IMAGE SCRIPT TOOLS ON INPUT DATA TYPE

Field of the Invention

This invention relates generally to workflow management wherein computer software is used to implement the automated control of multiple-step, non-real-time processes. More particularly, this invention creates and executes custom processing flows by relying on a knowledge-based system rather than user understanding.

Background of the Invention

In many sophisticated product development processes, execution and oversight of the process is a significant portion of the production effort. Operators must be trained to understand production procedures and the use of individual tools. Supervisors must track 10 and schedule production tasks. File and disk space management has been estimated to consume as high as 80 percent of the production labor time.

Workflow management is used to automate production systems. Workflow management has two major sub-fields, control of standard practices and creation of custom processing flows. Workflow management of standard practices is commonly 15 applied to business processes, and generally consists of software in which one or more workflow procedures are embodied in the software. Workflow management of custom processing flows generally consists of systems to build procedures from a collection of tools. Such systems are commonly found in commercial image processing packages (Khoros, Erdas Imagine, AVS), and rely on the user to understand the processing and

- 2 -

tools being used. Such image processing systems do not generally support the status and tracking needed for production.

Alternatively, workflow management can be performed manually, using tools for tracking status. When done manually, however, operators must track and record the
5 location of all intermediate files, and production managers must control the movement of data between the different operating stages.

All of the existing methods for workflow management of custom processing flows require a great deal of knowledge and/or training on the part of the users. The user
must be aware of the availability, use, and caveats of each software tool to be
10 incorporated in the workflow procedure.

Depending upon the particular method being used, the user may also be responsible for tracking filenames of intermediate products, recording completion status information, and moving data between configured areas. The effort required to do this has been measured to be up to 40 percent of the total labor cost of the production.

15

Summary of the Invention

This invention was developed in response to the needs set forth above, but with the recognition that production processes are constantly evolving due to the introduction of new tools and new products. The resulting Process Automation System according to the invention automatically creates and executes workflow management procedures for a
20 variety of applications. Not only does the system handle lower-level tasks such as

- 3 -

bookkeeping and file shuffling, it can also be used to dynamically develop new production procedures and guide the operators through them once created.

Broadly, the effectiveness of the process is twofold, in that it automatically or semi-automatically creates the workflow management procedure, and executes the 5 procedure in an environment that provides the status, tracking and control needed for a production environment.

The system includes three main components:

- Data and Tool Definition;
- Work Order Creation; and
- 10 • Work Order Execution.

In the Data and Tool Definition component, software tools and their inputs and outputs are defined to the system via a graphical user interface. This provides the information that the system needs to reason over how to create a workflow management procedures and dynamically generated user interfaces.

15 In the Work Order Creation component, the creation of the workflow management procedure is initiated by the user identifying the desired output product and the desired input products. The user may also provide the name of a previously stored, abstracted template to provide guidance and eliminate the need for some interaction in the semi-automatic operational mode. The Work Order Creation component recursively 20 reasons backwards from the output product to the input products, determining what processing steps and intermediate products are needed to perform the processing. When multiple valid options exist, the user or the abstracted template is consulted for guidance.

- 4 -

All prompting for inputs is done via dynamically generated graphical user interfaces, based upon the information entered to the data and tool definition component. Once a valid processing path exists for a single input, that path can be automatically replicated for similar inputs.

5 Once created, Work Order Execution is invoked to execute the series of tasks, and to monitor and control status. The Work Order Execution component is largely based upon the WPEP Scheduler, built by Russell Sieron in 1996. Additional capabilities were created as part of this development that interface with the Process Automation System database and partition workflow management procedures into blocks of processing steps.

10

Brief Description of the Drawings

FIGURE 1 depicts the logical structure of the system when used with the graphical user interface to create and execute a workflow management procedure;

FIGURE 2 depicts the logical structure of the system when run without the graphical user interface for workflow management procedure creation;

15 FIGURE 3 shows a Product and Tools Registration step that provides a graphical user interface for various operations;

FIGURE 4 is set of diagrams which illustrate the atomic definition of tools;

FIGURE 5 shows how recursive reasoning may be used to build a workflow management procedure;

20 FIGURE 6 shows a workflow management procedure for a single input image; and

- 5 -

FIGURE 7 illustrates a workflow management procedure following replication of a branch.

Detailed Description of the Invention

The Process Automation System according to the invention provides a software utility for the creation and execution of workflow management procedures. It is useful in a multitude of applications, and is currently being applied in the field of image processing and geospatial information production (under the tradename GeoWorxTM). A current implementation consists of C and JAVA source code designed for use in the UNIX operating system environment. It is currently implemented for Solaris 2.6 on a SUN workstation. The system integrates the following subsystems:

1. Manual Work Order Creation – the user interface for creation and execution of workflow management procedures and abstracted templates (called profiles);
2. Event Driven Work Order Creation – a command-line interface for creation and execution of workflow management procedures;
3. Work Order Interpreter – the logic engine that reasons over needed inputs and outputs and available tools to create them, and creates the workflow management procedure;
4. Run Control – the runtime control engine that analyzes the workflow management procedure for parallelism and creates the commands to execute the tasks;

- 6 -

5. Scheduler – the process management control executive that oversees process execution on each host, provides real-time and logged status, and provides for user intervention to reschedule or terminate jobs;
 6. Database – a centralized database that holds the knowledge base for reasoning, catalogs of available input data, and processing logs;
 7. Task Support – utilities for conveying status information back to Run Control; and
 8. Product and Tool Registration – an application that provides a graphical user interface for registering new product types and tools in the system.
- 10 Among other differences to prior-art approaches, the Work Order Interpreter subsystem can be used to create a workflow management procedure by reasoning over the capabilities and dependencies of tools that perform specific tasks. Given the specification of the desired product and input(s), the Work Order Interpreter searches a database of known tools to determine the needed tools and their sequences to create the
- 15 product from the inputs. Supported workflow procedure topologies include linear sequences, trees, and lattices. The Work Order Interpreter can be run in fully automatic mode, in which default selections are always made, or in semi-automatic mode, in which the user is allowed to choose among the valid choices available. Workflow management procedures can be stored in an abstracted form and recalled for later use with new inputs,
- 20 even if the number of inputs has changed.

Figure 1 depicts the logical structure of the system when used with the graphical user interface to create and execute a workflow management procedure. The user

- 7 -

interacts with an application that utilizes the Manual Work Order Creation, Work Order Interpreter, and Database subsystems. The user selects the desired type of output product and the input data from which to make it using the graphical user interface of this application. The user may optionally select to use a pre-stored abstracted template 5 (profile) to provide further direction and defaults to the Work Order Interpreter.

The Work Order Interpreter then reasons through the creation of the workflow management procedure, stopping only to obtain inputs for information specific to the job (i.e. for what part of the Earth should the output product be made) or to resolve ambiguities if a profile is not provided. Once the potential workflow management 10 procedure has been completed, the user is allowed to review and modify the procedure as desired. The workflow management procedure is then stored in the database and the run control application is spawned.

The run control application obtains the workflow management procedure from the database and analyses it for dependencies and parallelism. As each group of tasks that 15 can be executed in parallel is reached, the control information in the workflow management procedure is converted into the proper commands to execute the task. The commands are then submitted to the scheduler for execution. The run control application is invisible to the user.

The scheduler manages the queue of submitted tasks on each host, ensuring that 20 no more than the allowed number of concurrent tasks are being executed. The scheduler provides status information about the queue via a graphical user interface. This interface allows the user to view the stderr and stdout output of tasks, postpone tasks, and

terminate tasks. Upon completion of each task, this stderr and stdout output is e-mailed to the user.

The task itself is run via a "task script," a shell script that acts as a wrapper to allow any application that can be run without a graphical user interface to be integrated 5 into the system. The task script utilizes a UNIX command, *StatusNotify*, to transmit status information directly back to the run control application. Any application that cannot be run without its graphical user interface can be run as a "Manual Application." When a manual application is found within a workflow management procedure, e-mail is sent to the individual who was specified to perform the task when the workflow 10 management procedure was created. The e-mail contains all of the instructions relevant to performing the task, such as the location of input files, the desired location of output files, and any special processing notes. The individual runs the graphical user interface to *StatusNotify* before processing begins. This confirms receipt of the message. The individual then performs the processing and runs *StatusNotify* to report the completion 15 status.

Figure 2 depicts the logical structure of the system when run without the graphical user interface for workflow management procedure creation. The system is accessed via a command-line front-end that can be spawned from another process. All needed information is provided on the command line. The remainder of the processing proceeds 20 as described above.

The Product and Tools Registration is used by software domain experts to register new product types and tools within the system. The logical flow of this capability is

depicted in Figure 3, which shows a Product and Tools Registration involving an application, *AddProcedures*, that provides a graphical user interface for all of these operations. A product is registered as a data type and the tools that are capable of creating that data type. A data type can also be registered to a data class, so that it can be
5 recognized by tools that work on a variety of input data types. Examples of products could include orthorectified images, mosaics, change detection images, and countless other imagery and non-imagery products. Tools are registered by providing the name of the task script to execute the tool, a categorization of how to invoke the tool (command line, manual, prompt-and-respond, etc.), and a detailed description of the inputs, outputs
10 and control parameters for the application. The language for making this description includes support for conditionally needed information, lists of information, multiplicity, and any permutation thereof. The user sees the language expressed as a tree, with graphical control of the tree layout and graphical user interface forms to provide the details for each tree node. In the process of defining the interface for the tool, the data
15 type or class of all inputs and outputs is provided. This provides the information over which the Work Order Interpreter reasons. Task scripts are created using a standard text editor.

The Work Order Interpreter views each tool (procedure) as an atomic unit that outputs a specific type or class of product when provided with specific types or classes of
20 inputs. The multiplicity of inputs and outputs is a characteristic of the tool. The Work Order Interpreter begins by considering the desired output product. It determines what tools are available to make the product. If multiple choices exist, it selects one based

- 10 -

upon either defaults, the abstracted template (profile) provided it, or user intervention. The method of selecting is based upon the settings of three flags to control its processing. The selected choice is then examined to determine its needed inputs. This processing continues recursively until either a specified input item is found to fulfill the needed 5 inputs, or it is determined that the product cannot be made from those inputs.

Additional ancillary inputs can be specified as needed. This processing is depicted in Figures 4 and 5. Once a processing flow has been established for a single input, then the remaining inputs are inspected to determine if they are of the same type. If one or more inputs of the same type exist, then the user is given the option of replicating the 10 processing flow branch for the corresponding inputs. When a processing flow branch is replicated, each processing step is evaluated to determine if it can accept additional inputs and outputs or if a new instance of that processing step must be created. This processing is depicted in Figures 6 and 7. The resulting workflow management procedure can be a single sequence, a tree, a lattice, or any permutation of the preceding.

15 Some advantages of the Process Automation System described herein include:

- Automated reasoning guides the user through creating the workflow management procedure, greatly reducing the need for training and expert operators;
- Abstracted templates can store workflow management procedures for reuse, recalling preferences and processing structure;
- 20 • Abstracted templates are not based upon the number of inputs, they are dynamically reconfigured to match the number of inputs each time they are used;

- 11 -

- Command-line software tools can be given a graphical user interface without any modification, recompiling, or relinking of the tool;
- The system can automate the use of any software tool that does not require input through a graphical user interface, including operating system commands and tools from other vendors;
- Inherently manual processes may be integrated into automated processing flows by sending e-mail instructions to operators;
- Automated execution eliminates file handling and tracking by operators; and
- Status and tracking capabilities support the needs of production systems.

10 We claim:

- 12 -

1. A method of creating a workflow procedure, comprising the steps of:
 - 2 defining software tools available to the procedure;
 - 4 user specifying desired input and output products based upon the available tools;
 - 6 recursively reasoning backwards from the output products to the input products to construct a topology of tasks and intermediate products required for the procedure; and
2. The method of claim 1, further including the step of recalling a previously stored, abstracted template to assist in determining what tasks and intermediate products are needed to perform the procedure.
3. The method of claim 1, including tools featuring a mix of single and multiple inputs, such that the topology is linear, tree-like, a lattice, or any permutation thereof.
4. The method of claim 1, further including the step of prompting a user to intervene to assist in determining what tasks and intermediate products are needed to perform the procedure.
5. The method of claim 1, further including the step of automatically replicating topology segments used to process a single input to satisfy multiple input tasks.

- 13 -

6. The method of claim 1, further including the step of wrapping any input
2 that can be executed without a graphical user interface in a shell script to eliminate the
need for recompiling, relinking, or otherwise altering the input.

7. The method of claim 1, further including the step of treating an input that
2 requires a graphical user interface as a manual tool to eliminate the need for recompiling,
relinking, or otherwise altering the input.

8. The method of claim 1, further including the step of pre-validating user
2 specifications, such that only valid tasks and intermediate products are generated for the
procedure.

9. The method of claim 1, further including the step of identifying the data
2 products as data types and data classes for step of reasoning over input data requirements.

10. The method of claim 1, further including the step of identifying certain
2 tools as accepting items of a data class, and outputting items that belong to a data class, to
allow for the efficient description of general-purpose tools.

11. The method of claim 1, further including the step of representing the user
2 interface as a tree structure.

- 14 -

12. The method of claim 11, wherein the nodes of the tree may represent
2 conditional nodes that select one of the underlying branches.

13. The method of claim 11, wherein the nodes of the tree may represent
2 multiple occurrences of that item.

14. The method of 13, wherein the number of occurrences of an item may be
2 tied to an arbitrary range or the number of occurrences of another item.

15. A system for creating a workflow procedure, comprising:
2 a data and tool definition component for defining available software tools along
with their inputs and outputs;
4 a work order creation component enabling a user to specify desired input and
output products using the data and available tools, the work order creation component
6 being operative to recursively reason backwards using the output products to the input
products to create a topology of tasks and intermediate products needed to perform the
8 workflow procedure; and
10 a work order execution component operative to execute the tasks, and monitor the
status of the procedure.

- 15 -

16. The system of claim 15, further including a database for storing abstracted
2 templates to assist in determining the tasks and intermediate products needed to perform
the procedure.

17. The method of claim 15, wherein tools featuring a mix of single and
2 multiple inputs, such that the topology is linear, tree-like, a lattice, or any permutation
thereof.

18. The method of claim 15, further including a graphical interface for
2 prompting a user to assist in determining what tasks and intermediate products are needed
to perform the procedure.

19. The method of claim 15, wherein the work order creation component is
2 further operative to automatically replicate topology segments used to process a single
input to satisfy multiple input tasks.

FIGURE 1

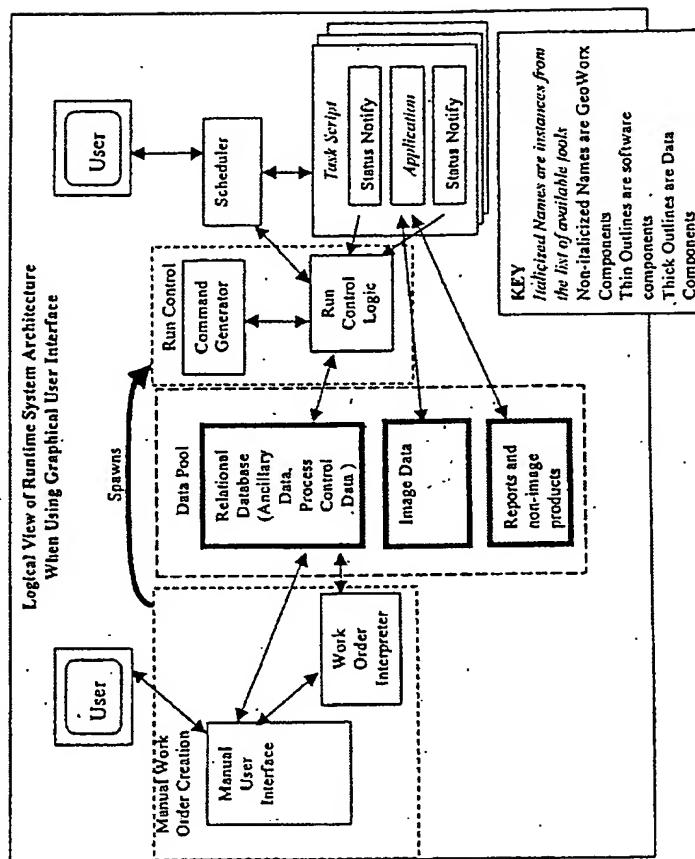


FIGURE 2

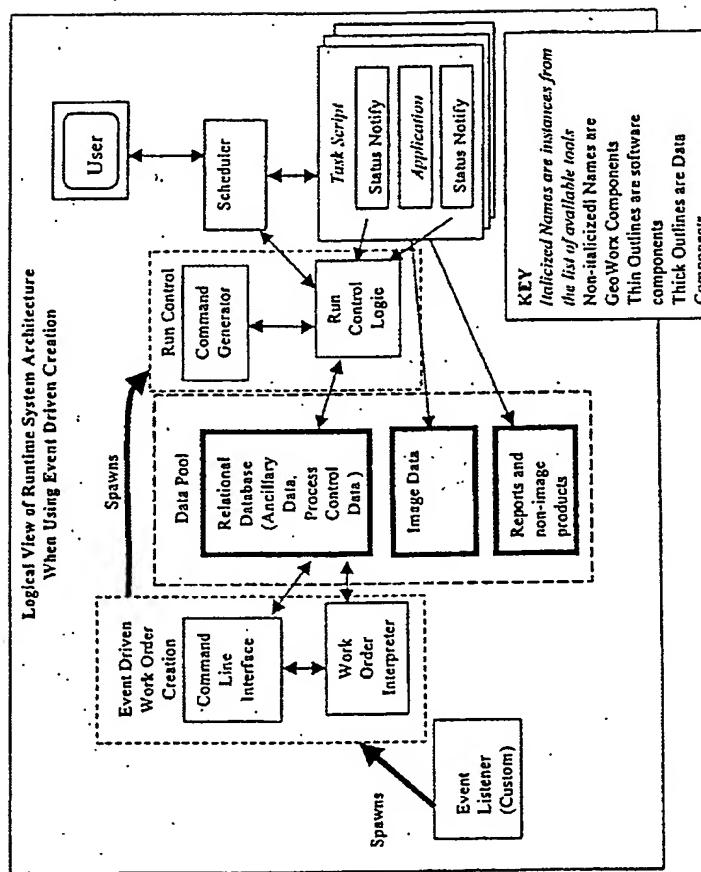


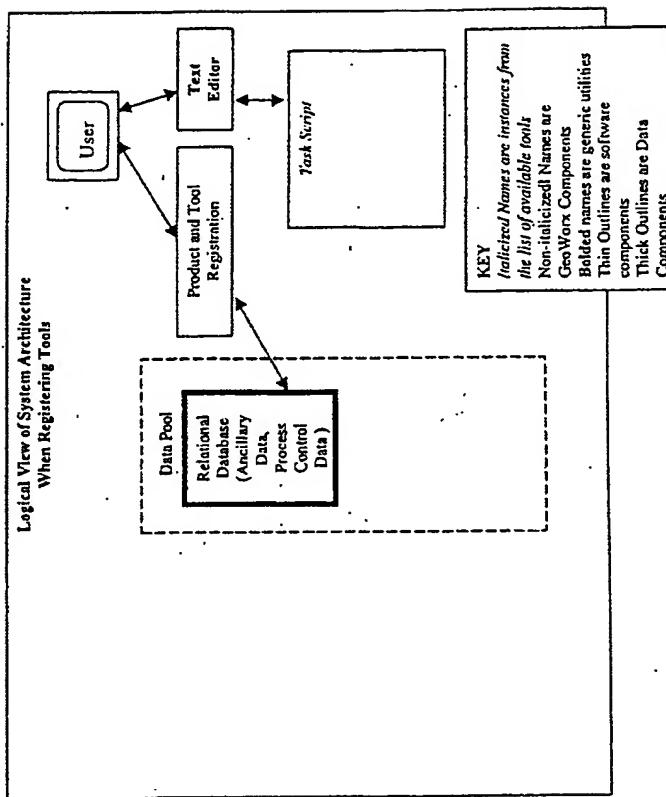
FIGURE 3

FIGURE 4

Atomic Definitions of Tools

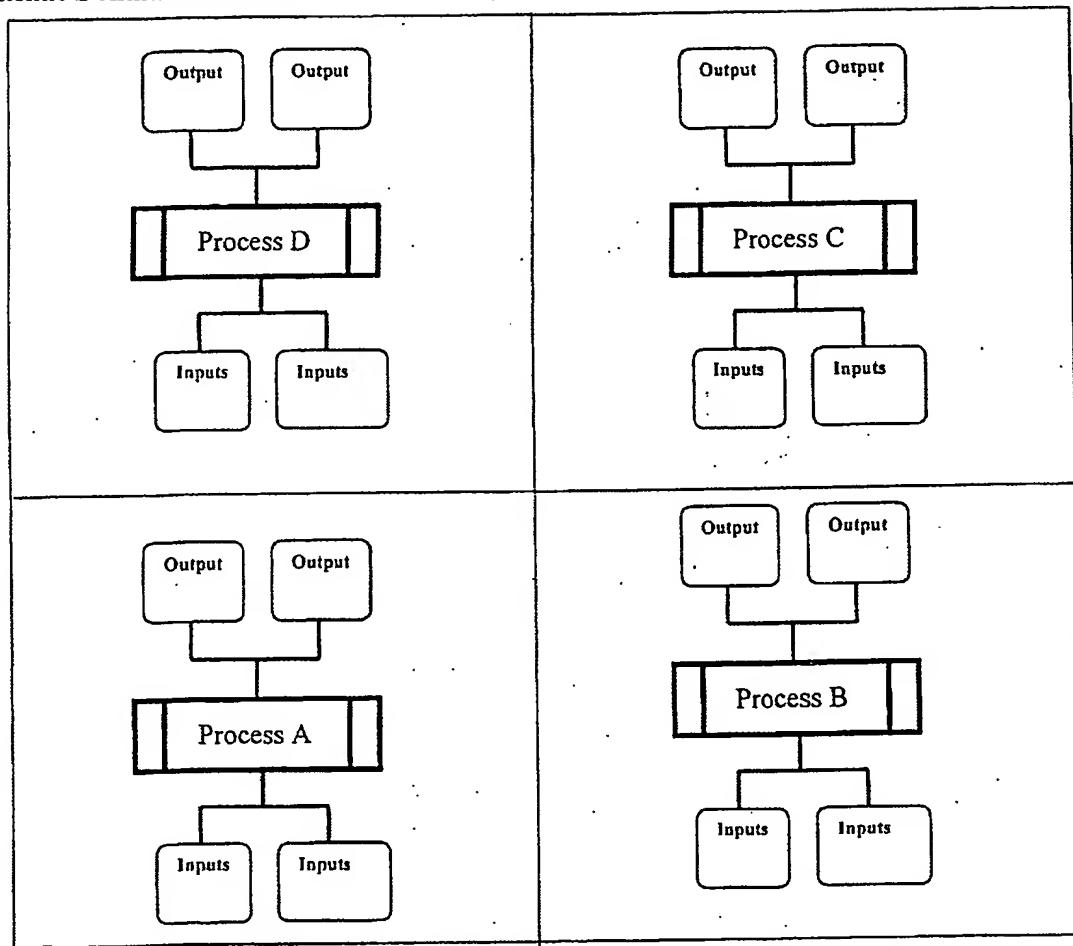


FIGURE 5
Recursive Reasoning Builds Workflow Management Procedure

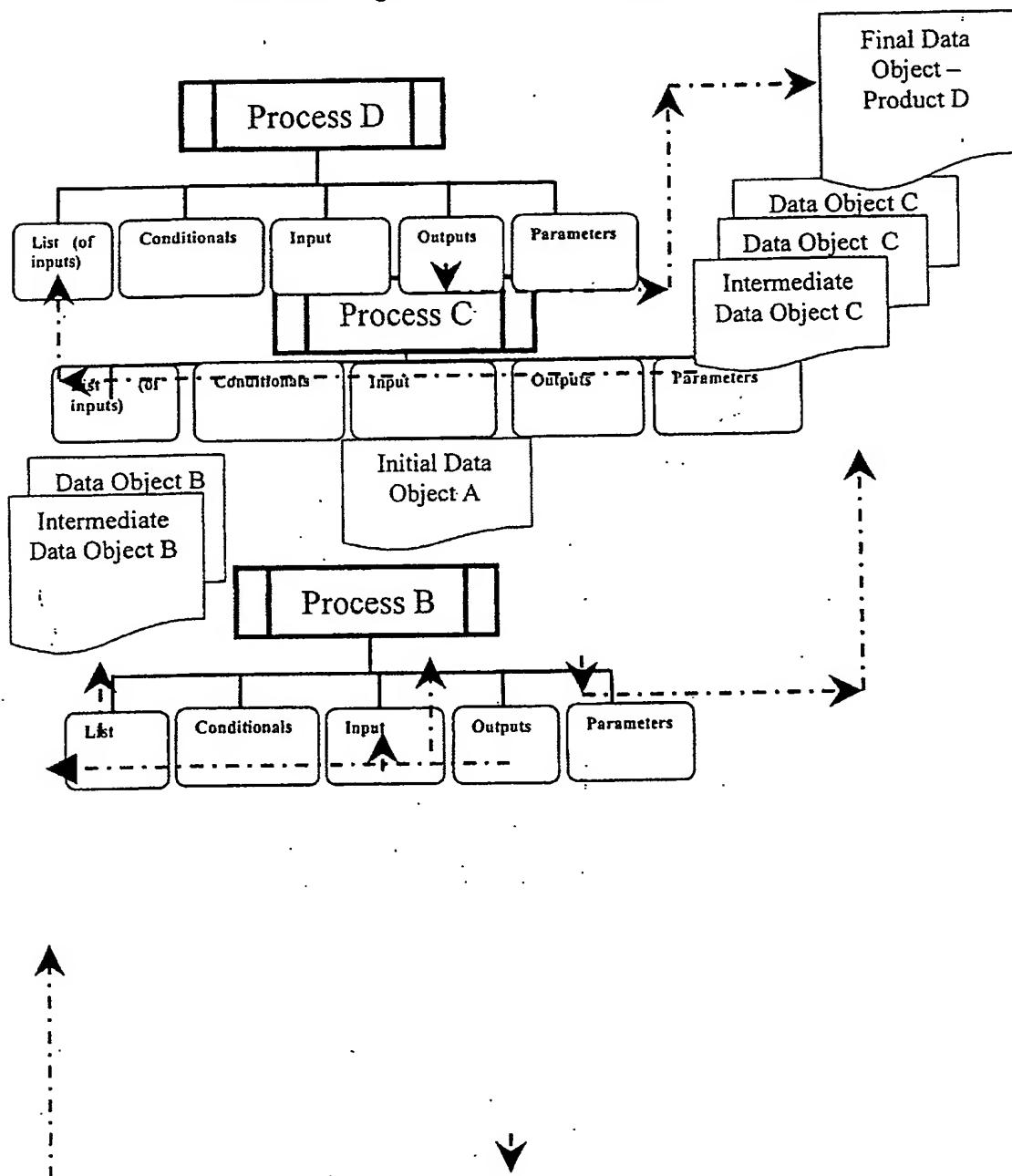


FIGURE 6

Workflow Management Procedure Branch for a Single Input Image

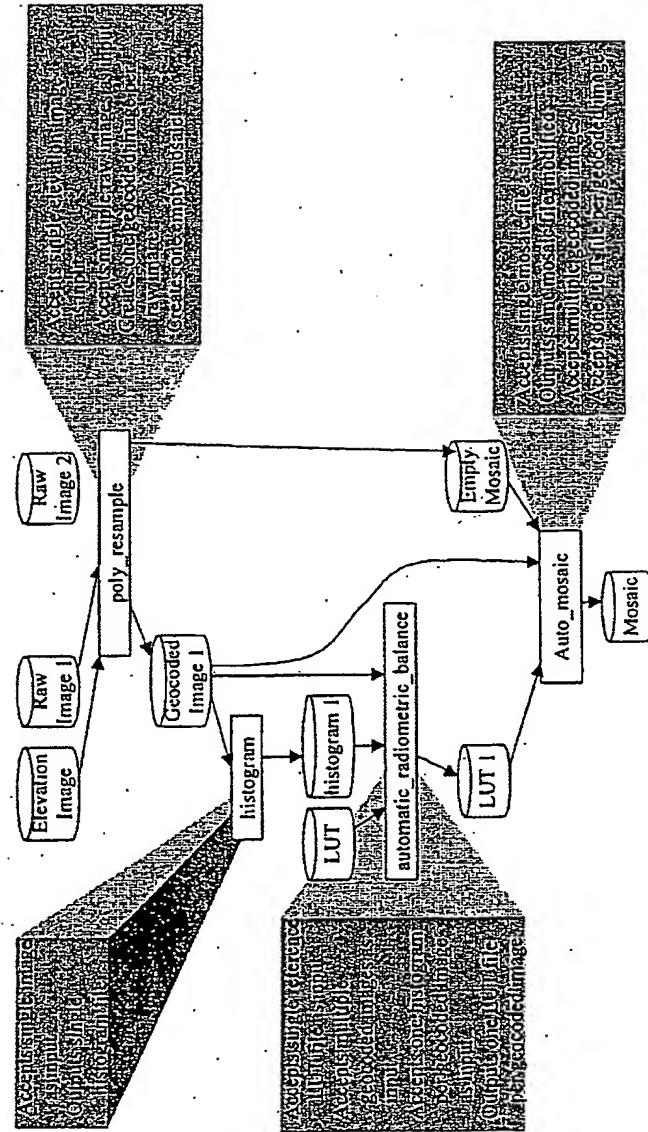
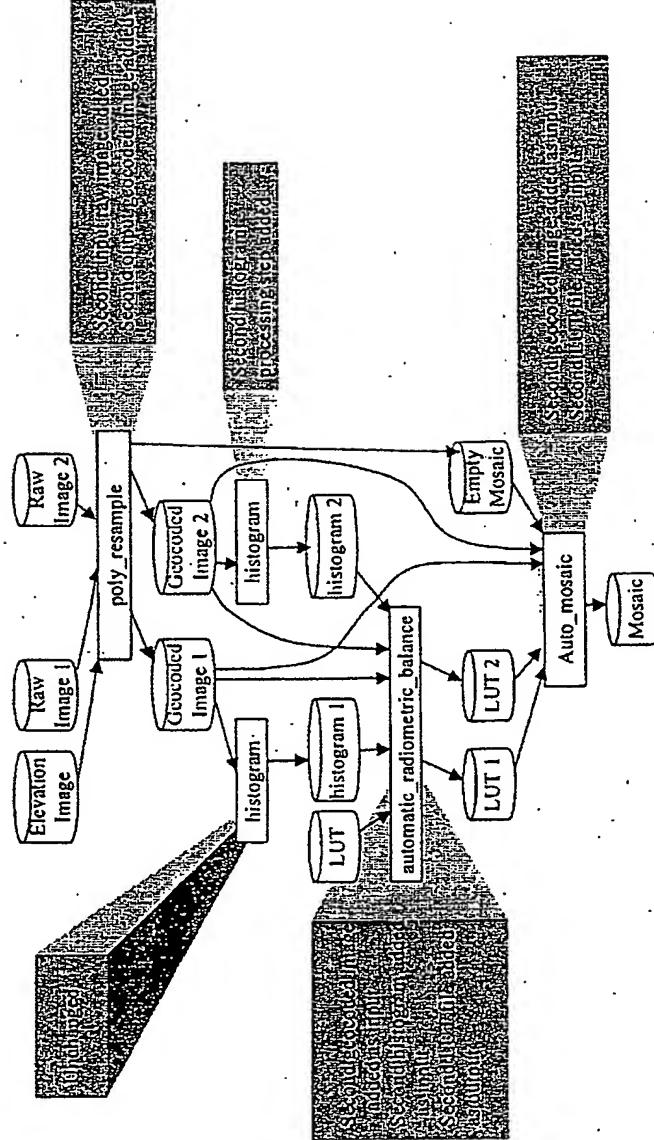


FIGURE 7**Workflow Management Procedure After Replication of Branch****BEST AVAILABLE COPY**

INTERNATIONAL SEARCH REPORT

International application No.
PCT/US01/05712

A. CLASSIFICATION OF SUBJECT MATTER

IPC(7) : G06F 17/60
 US CL : 705/8,9

According to International Patent Classification (IPC) or to both national classification and IPC

B. FIELDS SEARCHED

Minimum documentation searched (classification system followed by classification symbols)
 U.S. : 705/8,9

Documentation searched other than minimum documentation to the extent that such documents are included in the fields searched

Electronic data base consulted during the international search (name of data base and, where practicable, search terms used)
 EAST (USPAT, Derwent, IBMDB), ACM (Association of Computing Machinery online database)

C. DOCUMENTS CONSIDERED TO BE RELEVANT

Category *	Citation of document, with indication, where appropriate, of the relevant passages	Relevant to claim No.
X	US 5,737,727 A (LEHMANN et al) 07 April 1998 (07.04.1998).	1,15
Y		2, 4, 6-8, 10-12,16,18
Y	JORDAN, R. et al. cantata: A Data Flow Visual Language. 1995, downloaded April 4, 2001 from www.cs.ioc.ee/~khoros2/k2tools/cantata/cantata.html.	2, 4, 6-8, 10-12,16,18
Y	WOOD, Jason et al. Collaborative Visualization. Proceedings of the conference on Visualization, October 18-24, 1997.	2, 4, 6-8, 10-12,16,18

Further documents are listed in the continuation of Box C.

See patent family annex.

* Special categories of cited documents:

- "A" document defining the general state of the art which is not considered to be of particular relevance
- "E" earlier application or patent published on or after the international filing date
- "L" document which may throw doubts on priority claim(s) or which is cited to establish the publication date of another citation or other special reason (as specified)
- "O" document referring to an oral disclosure, use, exhibition or other means
- "P" document published prior to the international filing date but later than the priority date claimed

"T"

later document published after the international filing date or priority date and not in conflict with the application but cited to understand the principle or theory underlying the invention

"X"

document of particular relevance; the claimed invention cannot be considered novel or cannot be considered to involve an inventive step when the document is taken alone

"Y"

document of particular relevance; the claimed invention cannot be considered to involve an inventive step when the document is combined with one or more other such documents, such combination being obvious to a person skilled in the art

"&"

document member of the same patent family

Date of the actual completion of the international search

24 April 2001 (24.04.2001)

Date of mailing of the international search report

14 MAY 2001

Name and mailing address of the ISA/US

Commissioner of Patents and Trademarks
 Box PCT
 Washington, D.C. 20231

Facsimile No. (703)305-3230

Authorized officer

Tariq Hafiz

Telephone No. 703.305.3900

INTERNATIONAL SEARCH REPORT

I	al application No.
PCT/US01/05712	

Box I Observations where certain claims were found unsearchable (Continuation of Item 1 of first sheet)

This international report has not been established in respect of certain claims under Article 17(2)(a) for the following reasons:

1. Claim Nos.: because they relate to subject matter not required to be searched by this Authority, namely:

2. Claim Nos.: 3,5,9,13,14,17,19 because they relate to parts of the international application that do not comply with the prescribed requirements to such an extent that no meaningful international search can be carried out, specifically:
Please See Continuation Sheet

3. Claim Nos.: because they are dependent claims and are not drafted in accordance with the second and third sentences of Rule 6.4(a).

Box II Observations where unity of invention is lacking (Continuation of Item 2 of first sheet)

This International Searching Authority found multiple inventions in this international application, as follows:

1. As all required additional search fees were timely paid by the applicant, this international search report covers all searchable claims.
2. As all searchable claims could be searched without effort justifying an additional fee, this Authority did not invite payment of any additional fee.
3. As only some of the required additional search fees were timely paid by the applicant, this international search report covers only those claims for which fees were paid, specifically claims Nos.:

4. No required additional search fees were timely paid by the applicant. Consequently, this international search report is restricted to the invention first mentioned in the claims; it is covered by claims Nos.:

Remark on Protest

<input type="checkbox"/>
<input type="checkbox"/>

The additional search fees were accompanied by the applicant's protest.

No protest accompanied the payment of additional search fees.

INTERNATIONAL SEARCH REPORT

International application No.

PCT/US01/05712

Continuation of Item 4 of the first sheet: WORKFLOW SYSTEM AND BUILDER OFFERS IMAGE SCRIPT TOOLS ON INPUT DATA TYPE

Continuation of Box I Reason 2: Claim 3 recites *including tools featuring a mix of single and multiple inputs, such that the topology is linear, tree-like, a lattice, or any permutation thereof.*

Claim 3 is unsearchable because 1) it is unclear how to apply the language of the claim to arrive at the number or inclusion of items in the permutations of the mix being claimed; 2) it is unclear as to what linear, tree-like, or lattice refers. The specification does not cure the aforementioned deficiencies.

Claim 5 recites the *step of automatically replicating topology segments.*

Claim 5 is unsearchable because it is unclear as to what *replicating topology segments* refers. The specification does not cure the deficiency.

Claim 9 recites the *step of identifying the data products as data types and data classes for step of reasoning over input data requirements.*

Claim 9 is unsearchable because it is unclear as to what *for step of reasoning over input data requirements* refers. There is no antecedent basis for this limitation. The specification does not cure the deficiency.

Claim 13 recites *wherein the nodes of the tree may represent multiple occurrences of that item.*

Claim 13 is unsearchable because 1) it is unclear as to what *the tree* is being referred to; and 2) it is unclear as to what *that item* refers to. There is no antecedent basis for this limitation. The specification does not cure the aforementioned deficiencies.

Claim 14 is unsearchable because claim 14 depends on claim 13.

Claim 17 recites *wherein tools featuring a mix of single and multiple inputs, such that the topology is linear, tree-like, a lattice, or any permutation thereof.*

Claim 17 is unsearchable because 1) it is unclear how to apply the language of the claim to arrive at the number or inclusion of items in the permutations of the mix being claimed; 2) it is unclear as to what linear, tree-like, or lattice refers; 3) it is unclear as to how *featuring* further limits the base claim. The specification does not cure the aforementioned deficiencies.

Claim 19 recites the *step of automatically replicating topology segments.*

Claim 19 is unsearchable because it is unclear as to what *replicating topology segments* refers. The specification does not cure the deficiency.